In the Specification:

Please replace paragraph [0004] with new paragraph [0004], shown below.

[0004] Most real-world software systems of any significant complexity are written in more than one programming language. For example, an environment may be implemented in Java while an interpreted language may be running on top of Java and need to be debugged. This situation creates significant difficulties for software developers attempting to debug these systems. This problem is complicated by the fact that there is no standardization in terms of internal structures, such as stack frames, between different programming languages. For example, it is not uncommon for a developer to see stack information not directly related to the software being debugged when encountering a stack frame for one language, when using a debugger intended for another language. As another example, when using a debugger intended for the Java language, a Java stack will not include the stack for XScript (a JavaScript variant with native support for extensible markup language (XML)), and can sometimes show the set of Java classes that implement the XScript engine (these are part of the environment, but not the software the developer is working on). One multi-language debugger, described in Java Specification Request (JSR) 45, can only be used to debug languages that are easily transformed into Java and then compiled. This and most other multi-language debuggers won't work with languages such as XScript that where the language will be run by an interpreter or the language can not be mapped directly to Java because, for example, the language has a different data structure. Thus, creating debugging tools that can be applied to software applied to more than one programming language, and running in the same environment, has proved to be extremely difficult.

Please replace the Abstract with the new Abstract shown below.

~~[0025]~~ Software developers working on multi-language systems can utilize a multi-language debugging environment. The debugging environment can be uniform across languages, and can seamlessly perform debugging between one or more languages in a multi-language environment. Such a system can have a number of attributes intended to help developers facing debugging problems in multi-language environments.

~~[0026] This description is not intended to be a complete description of, or limit the scope of, the invention. Other features, aspects, and objects of the invention can be obtained from a review of the specification, the figures, and the claims.~~